Optimization Problems in Model Predictive Control

Stephen Wright
Jim Rawlings, Matt Tenny, Gabriele Pannocchia
University of Wisconsin-Madison

FoCM '02
Minneapolis
August 6, 2002

reminder!

Send your new tech reports to

http://www.optimization-online.org

Tech report repository, sponsored by Math Programming Society.

themes

Industrial control is a rich source of optimization problems (also uses tools from control theory, PDE, linear algebra). Foundations *in* Computational Mathematics!

Real-time imperative makes efficient algorithms important.

Describe a feasible trust-region SQP method that is

- ▷ simple, yet with good convergence properties
- ▷ particularly well suited to the nonlinear MPC problem.

outline

- introduction to optimal control, model predictive control (MPC)
- linear MPC:
  - ▷ algorithms
  - ▷ near-optimal solutions for infinite-horizon problems
- a feasible SQP method
- nonlinear MPC
  - ▷ customizing the feasible SQP algorithm
  - ▷ computational results

control: introduction

Control problems consist of

- ▷ a dynamic process (“state equation”, “model”); and
- ▷ ways to influence evolution of that process (“controls” or “inputs”).

An engineer may want to

- ▷ steer the process toward some desired state, or operating range, or avoid some undesirable states;
- ▷ transition between two states in an optimal way;
- ▷ optimize some function of the process state and controls, or minimize time needed to reach some specified goal.

industrial control examples

- oil refining and petrochemicals
- chemicals
- food processing
- mining
- furnaces
- pulp and paper

state equation

$x(t) =$ state at time t; $u(t) =$ inputs at time t.
May not be able to measure the state x, only some observation $y = g(x)$.

State equation describes process evolution:

$$\dot{x} = F(x, u, t).$$

May be naturally an ODE, or possibly derived from a parabolic PDE. (Also may be generalized to DAE.)

Discretization in time leads to

$$x_{k+1} = F_k(x_k, u_k), \quad k = 0, 1, 2, \ldots.$$

The case of F (or F_k) linear is important: gives adequate performance in many cases, good algorithms and software available.

objective and constraints

Objectives are often simple, e.g. convex quadratic:

$$L(x,u) = \frac{1}{2}\sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k + \frac{1}{2} x_N^T \tilde{Q} x_N.$$

(Q symmetric positive semidefinite; R, Π symmetric positive definite.)

May have auxiliary constraints on states and controls. Examples:

desired operating range:	$L \le x_k \le U, \quad \forall k;$
actuator limits:	$h(u_k) \le 0, \quad \forall k;$
rate limits:	$-r \le u_{k+1} - u_k \le r, \quad \forall k.$

Can make "soft constraints" by including quadratic penalty in the objective; or impose explicitly as "hard constraints".

setpoints

In industrial applications often have a *setpoint* x_s describing the optimal steady state, with corresponding inputs u_s. (x_s, u_s) chosen to hit some target observation. Role of the controller is to steer the process to (x_s, u_s) and keep it there, despite disturbances.

Usually choose (x, u) to measure deviation from (x_s, u_s).

Often suffices to linearize the process dynamics around the setpoint, to obtain a linear, homogeneous model F:

$$x_{k+1} = F(x_k, u_k) = Ax_k + Bu_k.$$

open-loop (optimal) control

Given current state x_0, choose a time horizon N (long), and solve the optimization problem for $x = \{x_k\}_{k=0}^{N}$, $u = \{u_k\}_{k=0}^{N-1}$:

$$\min L(x, u), \quad \text{subject to } x_0 \text{ given,}$$

$$x_{k+1} = F_k(x_k, u_k), \quad k = 0, 1, \ldots, N-1, \quad \text{other constraints on } x, u.$$

Then apply controls $u_0, u_1, u_2, \ldots$.

- ▷ flexible with respect to nonlinearity and constraints;
- ▷ if model F_k is inaccurate, solution may be bad;
- ▷ doesn't account for system disturbances during time horizon;
- ▷ never used in industrial practice!

closed-loop (feedback) control

Determine a *control law* $K(\cdot)$ such that $u = K(x)$ is the optimal control setting to be applied when the current state is x.

To control the process, simply measure the state x at each timepoint, calculate and apply $u = K(x)$.

- ▷ for special cases (quadratic objective, linear state equation) K is a linear function, calculated by solving a Riccati equation;
- ▷ more robust with respect to model error;
- ▷ feedback: responds to disturbances;
- ▷ *very* difficult to find K when model nonlinear or has constraints;
- ▷ ad-hoc methods for handling constraints (e.g. clipping) not reliable.

model predictive control (MPC)

Given current state x_0, time horizon N, solve the optimization problem:

$$\min L(x,u), \quad \text{subject to } x_0 \text{ given,}$$
$$x_{k+1} = F_k(x_k, u_k), \quad k = 0, 1, \ldots, N-1, \quad \text{other constraints on } x, u.$$

Then apply control u_0. At next timepoint $k = 1$, estimate the state, and define a new N-stage problem starting at the current time (*moving horizon*). **Repeat** indefinitely.

- ▷ performs closed-loop control using open-loop techniques; retains advantages of each approach;
- ▷ use state/control profile at one timepoint as basis for a starting point at the next timepoint;
- ▷ requires problem to be solved quickly (between timepoints).

other issues in MPC

- State estimation: Given observations y_k and inputs u_k, estimate the states x_k.
- Nominal stability: Assuming that the state equation is exact, can we steer the system to the desired state (usually $x = 0$) while respecting the constraints?
- Disturbance modeling: detecting and estimating disturbances and mismatches between model and actual process.

linear-quadratic regulator

Simplest control problem is, for given x_0:

$$\min_{x,u} \Phi(x,u) \stackrel{\text{def}}{=} \frac{1}{2}\sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k \quad \text{s.t.} \quad x_{k+1} = Ax_k + Bu_k.$$

From KKT conditions, dependence of optimal values of $x_1, x_2, \ldots$ and $u_0, u_1, \ldots$ on initial x_0 is linear, so have

$$\Phi(x,u) = \frac{1}{2} x_0^T \Pi x_0$$

for some s.p.d. matrix Π.

By using this dynamic programming principle, isolating the first stage, can write the problem as:

$$\min_{x_1,u_0} \frac{1}{2}(x_0^T Q x_0 + u_0^T R u_0) + \frac{1}{2} x_1^T \Pi x_1 \quad \text{s.t.} \quad x_1 = Ax_0 + Bu_0.$$

By substituting for x_1, get unconstrained quadratic problem in u_0. Minimizer is

$$u_0 = Kx_0, \quad \text{where} \quad K = -(R + B^T \Pi B)^{-1} B^T \Pi A.$$

so that

$$x_1 = Ax_0 + Bu_0 = (A + BK)x_0.$$

By substituting for u_0 and x_1 in

$$\tfrac{1}{2} x_0^T \Pi x_0 = \tfrac{1}{2}(x_0^T Q x_0 + u_0^T R u_0) + \tfrac{1}{2} x_1^T \Pi x_1,$$

obtain the *Riccati equation*:

$$\Pi = Q + A^T \Pi A - A^T \Pi^T B (R + B^T \Pi B)^{-1} B^T \Pi A.$$

There are well-known techniques to solve this equation for Π , hence K.

Hence, we have a feedback control law $u = Kx$ that is optimal for the LQR problem.

linear MPC

More general linear-quadratic problem includes constraints:

$$\min_{x,u} \frac{1}{2}\sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k, \quad \text{subject to}$$
$$x_{k+1} = Ax_k + Bu_k, \quad k = 0, 1, 2, \dots,$$
$$x_k \in X, \quad u_k \in U,$$

possibly also mixed constraints, and constraints on $u_{k+1} - u_k$.

Assuming that $0 \in \text{int}(X)$, $0 \in \text{int}(U)$ and that the system is stabilizable, we expect that $u_k \to 0$ and $x_k \to 0$ as $k \to \infty$. Therefore, for large enough k, the non-model constraints become inactive.

Hence, for N large enough, the problem is equivalent to the following (finite) problem:

$$\min_{x,u} \frac{1}{2}\sum_{k=0}^{N-1} x_k^T Q x_k + u_k^T R u_k + \frac{1}{2} x_N^T \Pi x_N, \quad \text{subject to}$$
$$x_{k+1} = A x_k + B u_k, \quad k = 0, 1, 2, \ldots, N-1$$
$$x_k \in X, \quad u_k \in U, \quad k = 0, 1, 2, \ldots, N-1,$$

where Π is the solution of the Riccati equation. In the "tail" of the sequence ($k > N$) simply apply the unconstrained control law.

(Rawlings, Muske, Scokaert, ...)

When constraints are linear, it remains to solve a (finite) convex, structured quadratic program.

details: interior-point method

(Rao, Wright, Rawlings). Solve

$$\min_{u,x,\epsilon} \sum_{k=0}^{N-1} \frac{1}{2}(x_k^T Q x_k + u_k^T R u_k + 2x_k^T M u_k + \epsilon_k^T Z \epsilon_k) + z^T \epsilon_k + x_N^T \Pi x_N,$$

subject to

$$\begin{array}{rcll}
x_0 & = & \widehat{x}_j, & \text{(fixed)} \\
x_{k+1} & = & Ax_k + Bu_k, & k = 0, 1, \ldots, N-1, \\
Du_k - Gx_k & \leq & d, & k = 0, 1, \ldots, N-1, \\
Hx_k - \epsilon_k & \leq & h, & k = 1, 2, \ldots, N, \\
\epsilon_k & \geq & 0, & k = 1, 2, \ldots, N, \\
Fx_N & = & 0. &
\end{array}$$

Introduce dual variables, use stagewise ordering. Primal-dual interior-point method yields a block-banded system at each iteration:

$$
\begin{bmatrix}
\cdots & Q & M & -G^T & A^T & & & & & \\
& M^T & R & D^T & B^T & & & & & \\
& -G & D & -\Sigma_k^D & & & & & & \\
& A & B & & & & & & -I & \\
& & & & & -\Sigma_{k+1}^{\epsilon} & & -I & & \\
& & & & & & -\Sigma_{k+1}^{H} & -I & H & \\
& & & & & -I & -I & Z & & \\
& & & & -I & & H^T & & Q & \cdots
\end{bmatrix}
\begin{bmatrix}
\vdots \\ \Delta x_k \\ \Delta u_k \\ \Delta \lambda_k \\ \Delta p_{k+1} \\ \Delta \xi_{k+1} \\ \Delta \eta_{k+1} \\ \Delta \epsilon_{k+1} \\ \Delta x_{k+1} \\ \vdots
\end{bmatrix}
=
\begin{bmatrix}
\vdots \\ r_k^x \\ r_k^u \\ r_k^\lambda \\ r_{k+1}^p \\ r_{k+1}^\xi \\ r_{k+1}^\eta \\ r_{k+1}^\epsilon \\ r_{k+1}^x \\ \vdots
\end{bmatrix}
$$

where Σ_k^D, Σ_{k+1}^ϵ, etc are diagonal.

By performing block elimination, get reduced system

$$
\begin{bmatrix}
R_0 & B^T & & & & & & & & & \\
B & & -I & & & & & & & & \\
& -I & Q_1 & M_1 & A^T & & & & & & \\
& & M_1^T & R_1 & B^T & & & & & & \\
& & A & B & & -I & & & & & \\
& & & & -I & Q_2 & M_2 & A^T & & & \\
& & & & & M_2^T & R_2 & B^T & & & \\
& & & & & A & B & \ddots & \ddots & & \\
& & & & & & & \ddots & Q_N & F^T \\
& & & & & & & & F &
\end{bmatrix}
\begin{bmatrix}
\Delta u_0 \\ \Delta p_0 \\ \Delta x_1 \\ \Delta u_1 \\ \Delta p_1 \\ \Delta x_2 \\ \Delta u_2 \\ \vdots \\ \Delta x_N \\ \Delta \beta
\end{bmatrix}
=
\begin{bmatrix}
\tilde{r}_0^u \\ \tilde{r}_0^p \\ \tilde{r}_1^x \\ \tilde{r}_1^u \\ \tilde{r}_1^p \\ \tilde{r}_2^x \\ \tilde{r}_2^u \\ \vdots \\ \tilde{r}_N^x \\ r^\beta
\end{bmatrix},
$$

which has the same structure as the KKT system of a problem without side constraints (soft or hard).

Can solve by applying a banded linear solver: $O(N)$ operations. Alternatively, seek matrices Π_k and vectors π_k such that the following relationship is satisfied between $\widehat{\Delta p}_{k-1}$ and $\widehat{\Delta x}_k$:

$$-\widehat{\Delta p}_{k-1} + \Pi_k \widehat{\Delta x}_k = \pi_k, \qquad k = N, N-1, \ldots, 1.$$

By substituting in the linear system, find a recurrence relation:

$$\Pi_N = \bar{Q}_N, \qquad \pi_N = \tilde{r}_N^x,$$

$$\begin{aligned}
\Pi_{k-1} &= Q_{k-1} + A^T \Pi_k A - \\
&\qquad (A^T \Pi_k B + M_{k-1})(R_{k-1} + B^T \Pi_k B)^{-1}(B^T \Pi_k A + M_{k-1}^T), \\
\pi_{k-1} &= \tilde{r}_{k-1}^x + A^T \Pi_k \tilde{r}_{k-1}^p + A^T \pi_k - \\
&\qquad (A^T \Pi_k B + M_{k-1})(R_{k-1} + B^T \Pi_k B)^{-1}(\tilde{r}_{k-1}^u + B^T \Pi_k \tilde{r}_{k-1}^p + B^T \pi_k).
\end{aligned}$$

The recurrence for Π_k is the discrete time-varying Riccati equation!

choosing the horizon N

If $(x, u) = (0, 0)$ is in the relative interior of the constraint set, can find N large enough to make the finite-horizon problem equivalent to infinite-horizon problem: e.g. successive doubling of N.

However if $(0, 0)$ is feasible but not in the relative interior, there may be no N with this property. This case arises often e.g. it may be reasonable to have an input valve fully open at some timepoints.

How can we choose N so that the finite-horizon problem (and its solution) approximates the corresponding infinite-horizon problem to a specified level of accuracy?

problem formulation

$$\mathcal{O}: \quad \min_{\{x_k,u_k\}_{k=0}^{\infty}} \frac{1}{2}\sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k, \qquad \text{subject to}$$
$$x_0 = \text{given}, \qquad x_{k+1} = Ax_k + Bu_k, \quad k = 0,1,2,\ldots,$$
$$Du_k \le d, \quad k = 0,1,2,\ldots,$$
$$Ex_k \le e, \quad k = 0,1,2,\ldots.$$

Since $(x,u) = (0,0)$ is feasible, we must have $d \ge 0$, $e \ge 0$. We assume in fact that $e > 0$, since otherwise arbitrarily small disturbances render the problem infeasible.

Seek upper and lower bounding problems with finitely many variables.

upper bounding problem

Denote by $\bar{D}$ the row submatrix of D corresponding to right-hand side components of zero. For the problem

$$\mathcal{U}(N): \quad \min_{\{x_k,u_k\}_{k=0}^{\infty}} \frac{1}{2}\sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k, \qquad \text{subject to}$$

$$x_0 = \text{given}, \qquad x_{k+1} = Ax_k + Bu_k, \quad k = 0, 1, 2, \ldots,$$

$$Du_k \le d, \quad k = 0, 1, 2, \ldots,$$

$$Ex_k \le e, \quad k = 0, 1, 2, \ldots,$$

$$\bar{D}u_k = 0, \quad k = N, N+1, \ldots,$$

we have under the usual assumptions that the constraints other than $\bar{D}$ are strictly satisfied for all N sufficiently large.

By a change of variables in u_k (to the null space of $\bar{D}$, can solve Riccati equation to find a cost-to-go matrix $\bar{\Pi}$ such that

$$x_N^T \bar{\Pi} x_N = \sum_{k=N}^{\infty} x_k^T Q x_k + u_k^T R u_k, \quad \text{subject to}$$
$$x_{k+1} = Ax_k + Bu_k, \quad \bar{D}u_k = 0, \; k = N, N+1, \ldots,$$

for any x_N. Hence, can rewrite $\mathcal{U}(N)$ as

$$\mathcal{U}(N) : \quad \min_{\{x_k, u_k\}_{k=0}^{\infty}} \tfrac{1}{2} \sum_{k=0}^{N-1} (x_k^T Q x_k + u_k^T R u_k) + \tfrac{1}{2} x_N^T \bar{\Pi} x_N, \; \text{ s. t.}$$
$$x_0 = \text{given}, \quad x_{k+1} = Ax_k + Bu_k, \quad k = 0, 1, 2, \ldots, N-1$$
$$Du_k \le d, \quad k = 0, 1, \ldots, N-1$$
$$Ex_k \le e, \quad k = 0, 1, \ldots, N.$$

For N sufficiently large, we have

$$\Phi_{\mathcal{U}}(N) \ge \Phi * .$$

lower bounding problem

Enforce side constraints only over a finite horizon:

$$\mathcal{L}(N): \quad \min_{\{x_k,u_k\}_{k=0}^{\infty}} \tfrac{1}{2}\textstyle\sum_{k=0}^{\infty} x_k^T Q x_k + u_k^T R u_k, \qquad \text{subject to}$$

$$x_0 = \text{given}, \qquad x_{k+1} = Ax_k + Bu_k, \quad k = 0, 1, 2, \ldots,$$

$$Du_k \leq d, \quad k = 0, 1, 2, \ldots, N-1$$

$$Ex_k \leq e, \quad k = 0, 1, 2, \ldots, N.$$

Hence, can compute the usual cost-to-go matrix Π, and obtain the following finite formulation:

$$\mathcal{L}(N): \quad \min_{\{x_k,u_k\}_{k=0}^{\infty}} \tfrac{1}{2}\textstyle\sum_{k=0}^{\infty}(x_k^T Q x_k + u_k^T R u_k) + \tfrac{1}{2}x_N^T \Pi x_N, \quad \text{s.t.}$$

$$x_0 = \text{given}, \qquad x_{k+1} = Ax_k + Bu_k, \quad k = 0, 1, \ldots, N-1$$

$$Du_k \leq d, \quad k = 0, 1, \ldots, N-1$$

$$Ex_k \leq e, \quad k = 0, 1, \ldots, N.$$

analysis

For all N sufficiently large, we have

$$\Phi_{\mathcal{L}}(N) \le \Phi* \le \Phi_{\mathcal{U}}(N).$$

By working in the right space (ℓ^2), we have that

$$\Phi_{\mathcal{L}}(N) \uparrow \Phi*, \qquad \Phi_{\mathcal{U}}(N) \downarrow \Phi*,$$

and that the solutions of the lower-bounding and upper-bounding problems also converge to the solution of the true problem, in the ℓ^2 norm.

By choosing N large enough, can obtain a setting u_0 (from the upper-bounding problem) that produces an objective within a guaranteed level of optimality (measured by $\Phi_{\mathcal{U}}(N) - \Phi_{\mathcal{L}}(N)$).

nonlinear MPC: introduction

$$\min_{x,u} \sum_{k=0}^{N-1} \{C(x_k, u_k) + \Xi(\eta_k)\} + x_N^T P x_N + \Xi(\eta_N), \quad \text{subject to}$$

$$x_0 \text{ given}, \qquad x_{k+1} = F_k(x_k, u_k),$$

$$Du_k \le d, \quad \eta_k = \max(Gx_k - g, 0),$$

where Ξ is a convex quadratic (soft constraints).
Why use a nonlinear model?

- most applications have nonlinear dynamics: nonlinear rate laws, temperature dependence of rate constants, vapor/liquid therodynamic equilibrium;
- measured properties (hence objective terms) may be nonlinear functions of the state;
- a linear approximation may give seriously suboptimal control.

- nonlinear MPC is a structured nonlinear program, possibly with local optima;
- not used much in practice, partly due to lack of reliable algorithms and software, but use is growing (in chemicals, air and gas, polymers)
- structured SQP methods, dynamic programming (Newton-like) approaches, gradient projection, and recently interior-point methods have all been tried, often in the context of open-loop optimal control.
- because of the MPC context, a good starting point often is available, though not after an upset.

Our experience shows that there is considerable advantage to retaining feasibility. This leads us to consider a method of the *feasible SQP* type.

feasible trust-region SQP method

$$\min f(z) \text{ subject to } c(z) = 0, \; d(z) \le 0,$$

where $z \in R^n$, $f : R^n \to R$, $c : R^n \to R^m$, and $d : R^n \to R^r$ are smooth (twice cts diff) functions. Denote feasible set by $\mathcal{F}$.

From a feasible point z, obtain step Δz by solving trust-region SQP subproblem:

$$\min_{\Delta z} m(\Delta z) \stackrel{\text{def}}{=} \nabla f(z)^T \Delta z + \frac{1}{2} \Delta z^T H \Delta z \text{ subject to}$$
$$c(z) + \nabla c(z)^T \Delta z = 0, \; d(z) + \nabla d(z)^T \Delta z \le 0,$$
$$\|D \Delta z\|_p \le \Delta$$

for some Hessian approximation H, scaling matrix D, trust-region radius Δ, and $p = 1, 2$, or ∞. **Subproblem is always feasible!**

feasibility perturbation

In general $z + \Delta z \notin \mathcal{F}$, except for important special case: linear constraints. Find a perturbed step $\widetilde{\Delta z}$ such that

- feasibility: $z + \widetilde{\Delta z} \in \mathcal{F}$;
- asymptotic exactness:

$$\|\Delta z - \widetilde{\Delta z}\| \leq \phi(\|\Delta z\|)\|\Delta z\|,$$

  where $\phi : R^+ \rightarrow R^+$ continuous, monotonically increasing with $\phi(0) = 0$.

If $\widetilde{\Delta z}$ with these properties cannot be found, decrease Δ and recalculate.

FP-SQP outline

Decide whether or not to take step using actual/predicted decrease ratio ρ_k defined by

$$\rho_k = \frac{f(z^k) - f(z^k + \widetilde{\Delta z}^k)}{-m_k(\Delta z^k)},$$

i.e. use f itself as the merit function.

Other aspects of the algorithm are identical to standard trust-region approach.

Given starting point z_0, trust-region upper bound $\bar{\Delta} \geq 1$, initial radius $\Delta_0 \in (0, \bar{\Delta}]$,

$\eta \in [0, 1/4)$, and $p \in [1, \infty]$;

for $k = 0, 1, 2, \cdots$

Obtain Δz^k, seek $\widetilde{\Delta z}^k$ with desired properties;

if no such $\widetilde{\Delta z}^k$ is found;

$\Delta_{k+1} \leftarrow (1/2)\|D_k \Delta z^k\|_p$;

$z^{k+1} \leftarrow z^k$;

else

Calculate ρ_k;

if $\rho_k < 1/4$

$\Delta_{k+1} \leftarrow (1/2)\|D_k \Delta z^k\|_p$;

else **if** $\rho_k > 3/4$ and $\|D_k \Delta z^k\|_p = \Delta_k$

$\Delta_{k+1} \leftarrow \min(2\Delta_k, \bar{\Delta})$;

else

$\Delta_{k+1} \leftarrow \Delta_k$;

if $\rho_k > \eta$

$z^{k+1} \leftarrow z^k + \widetilde{\Delta z}^k$;

else

$z^{k+1} \leftarrow z^k$;

end (for).

assumptions for global convergence

1. For Δz satisfying the linearized constraints, have for some $\delta \in (0,1)$ and all scaling matrices D that

$$\delta^{-1}\|\Delta z\|_2 \le \|D\Delta z\|_p \le \delta\|\Delta z\|_2.$$

2. Bounded feasible level set, and f, c, d, smooth on an open ndb of this set.

3. Given any $\hat{z}$ in the level set, then for all z in some nbd of $\hat{z}$, we have

$$\min_{v \in \mathcal{F}} \|v - z\| \le \zeta \left(\|c(z)\| + \|[d(z)]_+\|\right),$$

for some constant ζ (Hoffmann property).

Can show (following Robinson) that (3) holds when MFCQ is satisfied at $\hat{z}$.

well definedness

Given assumptions 1, 2, 3, there is Δ_{def} such that for any z in the level set, a perturbed step $\widetilde{\Delta z}$ with the desired properties can be found whenever $\Delta \leq \Delta_{\mathrm{def}}$.

global convergence: technical results

MFCQ assumption at feasible z: $\nabla c(z)$ has full column rank, and there is v such that $\nabla c(z)^T v = 0$ and $v^T \nabla d_i(z) < 0$ for all active i.

Key role is played by the following “linear” subproblem:

CLP(z, τ):
$$\min_w \nabla f(z)^T w \text{ subject to}$$
$$c(z) + \nabla c(z)^T w = 0, \;\; d(z) + \nabla d(z)^T w \le 0, \;\; w^T w \le \tau^2.$$

Analogous to Cauchy point in analysis of TR algorithms for unconstrained optimization.

Can relate $m(\Delta z)$ to the optimal value of CLP$(z, \delta^{-1}\Delta)$.

Can show that optimal value of CLP(z, τ) is zero iff z is stationary, and for z in the neighborhood of a nonstationary point, optimal value of CLP(z, τ) is bounded away from 0.

global convergence

Result I: If Assumptions 1, 2, 3 hold, and all limit points satisfy MFCQ, and approximate Hessians satisfy

$$\|H_k\| \leq \sigma_0 + \sigma_1 k.$$

(Quasi-Newton Hessians often have this property.) Then at least one of the limit points is stationary.

Result II: If Assumptions 1, 2, 3 hold, and the approximate Hessians satisfy $\|H_k\| \leq \sigma$, then there cannot be a limit point at which MFCQ holds but the KKT conditions do not.

In other words, all limit points either are stationary or fail to satisfy MFCQ.

local convergence: assumptions

Suppose that $z^k \to z^*$, where z^* satisfies linear independence constraint qualification, strict complementarity, second-order sufficient conditions.

Also make additional assumptions on the algorithm:

- ▷ Have an estimate $\mathcal{W}_k$ of the active set, such that $\mathcal{W}_k = \mathcal{A}^*$ for all k sufficiently large, where $\mathcal{A}^* = \{i = 1, 2, \ldots, r \,|\, d_i(z^*) = 0\}$.
- ▷ Have Lagrange multiplier estimates μ^k (for equality constraints) and λ^k (for inequality constraints) such that $(\mu^k, \lambda^k) \to (\mu^*, \lambda^*)$
- ▷ Perturbed step satisfies:

$$\|\Delta z - \widetilde{\Delta z}\| = O(\|\Delta z\|^2),$$

$$d_i(z^k + \widetilde{\Delta z}^k) = d_i(z^k) + \nabla d_i(z^k)^T \Delta z^k, \quad \forall i \in \mathcal{W}_k.$$

discussion

Given good estimates (μ^k, λ^k), can find a good $\mathcal{W}_k$. Given good $\mathcal{W}_k$, can use least-squares estimation to find (μ^k, λ^k).

Finding both (μ^k, λ^k) and $\mathcal{W}_k$ simultaneously is trickier, but a practical scheme that alternates estimates of these two quantities would probably not be difficult to devise.

The condition on d_i, $i \in \mathcal{W}_k$ represents an explicit second-order correction. Can show that a projection technique produces $\widetilde{\Delta z}$ satisfying (3), provided the other assumptions hold.

local convergence result

Assume that H_k is Hessian of the Lagrangian:

$$H_k = \nabla^2_{zz}\mathcal{L}(z^k, \mu^k, \lambda^k)$$

that 2-norm trust region is used, and that the assumptions above hold. Then we have $\rho_k \longrightarrow 1$, and $\{z^k\}$ converges Q-quadratically to z^*.

Applying FP-SQP to nonlinear MPC

(Tenny, Wright, Rawlings)

$$\min_{x,u} \sum_{k=0}^{N-1} \{C(x_k, u_k) + \Xi(\eta_k)\} + x_N^T \Pi x_N + \Xi(\eta_N), \quad \text{subject to}$$

$$x_0 \text{ given}, \qquad x_{k+1} = F_k(x_k, u_k),$$

$$Du_k \le d, \quad \eta_k = \max(Gx_k - g, 0).$$

Issues in applying FP-SQP to this problem:

- feasibility perturbation (stabilized)
- trust-region scaling
- approximate Hessians in the SQP subproblem

SQP subproblem for nonlinear MPC

$$\min_{\Delta x, \Delta u, \Delta \eta} \frac{1}{2}\Delta u_0^T \tilde{R}_0 \Delta u_0 + r_0^T \Delta u_0 + \tag{1}$$

$$\sum_{k=1}^{N-1} \left\{ \frac{1}{2} \begin{bmatrix} \Delta x_k \\ \Delta u_k \end{bmatrix} \begin{bmatrix} \tilde{Q}_k & \tilde{M}_k \\ \tilde{M}_k^T & \tilde{R}_k \end{bmatrix} \begin{bmatrix} \Delta x_k \\ \Delta u_k \end{bmatrix} + \begin{bmatrix} q_k \\ r_k \end{bmatrix}^T \begin{bmatrix} \Delta x_k \\ \Delta u_k \end{bmatrix} \right\}$$

$$+\frac{1}{2}\Delta x_N^T \tilde{Q}_N \Delta x_N + q_N^T \Delta x_N + \sum_{k=1}^{N} \Xi(\eta_k + \Delta \eta_k)$$

subject to

$$\begin{aligned}
\Delta x_0 &= 0, && \text{(2a)} \\
\Delta x_{k+1} &= A_k \Delta x_k + B_k \Delta u_k, \quad k = 0, 1, \ldots, N-1, && \text{(2b)} \\
D(u_k + \Delta u_k) &\le d, \quad k = 0, 1, \ldots, N-1, && \text{(2c)} \\
G(x_k + \Delta x_k) - (\eta_k + \Delta \eta_k) &\le g, \quad k = 1, 2, \ldots, N, && \text{(2d)} \\
\eta_k + \Delta \eta_k &\ge 0, \quad k = 1, 2, \ldots, N, && \text{(2e)} \\
\|\Sigma_k \Delta u_k\|_\infty &\le \Delta, \quad k = 0, 1, \ldots, N-1. && \text{(2f)}
\end{aligned}$$

Trust region is applied only to the u components (since x and η are defined in terms of u).

feasibility perturbation

Naive approach: Set $\widetilde{\Delta u} = \Delta u$, then recover $\widetilde{\Delta x}$ and $\widetilde{\Delta \eta}$ from

$$
\begin{aligned}
x_{k+1} + \widetilde{\Delta x}_{k+1} &= F(x_k + \widetilde{\Delta x}_k, u_k + \Delta u_k), \quad k = 0, 1, \ldots, N-1, \\
\eta_k + \widetilde{\Delta \eta}_k &= \max\left(G(x_k + \widetilde{\Delta x}_k) - g, 0\right), \quad k = 1, 2, \ldots, N.
\end{aligned}
$$

Often works fine. However on problems that are open-loop unstable (i.e. "increasing" modes in the model equation at the setpoint), it results in divergence of $\|\widetilde{\Delta x}_k - \Delta x_k\|$ as k increases.

Introduce a stabilizing change of variables based on a feedback gain matrix K_k, $k = 1, 2, \ldots, N-1$. Set $\widetilde{\Delta u}_0 = \Delta u_0$, then set remaining $\widetilde{\Delta u}_k$ and $\widetilde{\Delta x}_k$ to satisfy:

$$
\begin{aligned}
\widetilde{\Delta x}_{k+1} &= F(x_k + \widetilde{\Delta x}_k, u_k + \widetilde{\Delta u}_k) - x_k, \quad k = 0, 1, \ldots, N-1, \\
\widetilde{\Delta u}_k &= \Delta u_k + K_k(\widetilde{\Delta x}_k - \Delta x_k), \quad k = 1, 2, \ldots, N-1.
\end{aligned}
$$

Choose K_k such that

$$|\text{eig}(A_k + B_k K_k)| \leq 1.$$

- pole placement;
- solve the LQR problem based on (A_k, B_k) separately for each k;
- solve the time-varying LQR problem to get a set of K_k's. Results in a discrete Riccati equation, like the one encountered earlier in discussion of linear MPC: for $k = N-1, N-2, \ldots, 1$:

$$\begin{aligned} K_k &= -\left(R_k + B_k^T \Pi_{k+1} B_k\right)^{-1} \left(M_k^T + B_k^T \Pi_{k+1} A_k\right) \\ \Pi_k &= Q_k + K_k^T R_k K_k + M_k K_k + K_k^T M_k^T + \\ &\qquad (A_k + B_k K_k)^T \Pi_{k+1} (A_k + B_k K_k)\,. \end{aligned}$$

clipping; asymptotic exactness

If state constraints present, solve for $\widehat{\Delta u}_k$:

$$\min_{\widehat{\Delta u}_k} (\widehat{\Delta u}_k - \widetilde{\Delta u}_k)^T R_k (\widehat{\Delta u}_k - \widetilde{\Delta u}_k) \;\text{ subject to }\; D(u_k + \widehat{\Delta u}_k) \leq d,$$

and replace $\widetilde{\Delta u}_k \leftarrow \widehat{\Delta u}_k$.

Using implicit function theorem, can show that asymptotic exactness holds. The matrices K_k have the effect of improving the condition number in the Jacobian of the parametrized linear system to be solved for $(\widetilde{\Delta u}, \widetilde{\Delta x})$.

trust-region scaling

In unconstrained trust-region algorithms, have subproblem

$$\min_{\Delta z} \nabla f(z)^T \Delta z + \frac{1}{2} \Delta z^T H \Delta z, \quad \text{subject to} \;\; \|D\Delta z\|_2 \le \Delta,$$

whose solution is

$$(H + \xi D^T D)\Delta z = -\nabla f(z),$$

for some $\xi \ge 0$. Often choose D diagonal, with $D_{ii} = \sqrt{H_{ii}}$, $i = 1, 2, \ldots, n$.

Look for a corresponding strategy here.

By eliminating Δx and $\Delta \eta$ components from the subproblem (using linear constraints), get subproblem objective of the form

$$\frac{1}{2}\Delta u^T \widehat{Q} \Delta u + \widehat{r}^T \Delta u,$$

where

$$\Delta u = (\Delta u_0, \Delta u_1, \ldots, \Delta u_{N-1})$$

Thus, it makes sense to define the scaling matrices Σ_k in the trust-region constraint $\|\Sigma_k \Delta u_k\|_\infty \leq \Delta$ in terms of the diagonal blocks of $\widehat{Q}$. Can show that $\widehat{Q}_{kk}$ can be obtained as follows: Define $\mathcal{G}_N = \tilde{Q}_N$, then

$$\mathcal{G}_{k-1} = \tilde{Q}_{k-1} + A_{k-1}^T \mathcal{G}_k A_{k-1}, \quad k = N, N-1, \ldots, 2.$$

Then have

$$\widehat{Q}_{kk} = \tilde{R}_{k-1} + B_{k-1}^T \mathcal{G}_k B_{k-1}, \;\; k = 1, 2, \ldots, N.$$

Hessian approximation in the QP subproblem

Recall that the Hessian is block diagonal:

$$
\begin{gathered}
\tfrac{1}{2}\Delta u_0^T \tilde{R}_0 \Delta u_0 + r_0^T \Delta u_0 + \\
\sum_{k=1}^{N-1} \left\{ \tfrac{1}{2} \begin{bmatrix} \Delta x_k \\ \Delta u_k \end{bmatrix} \begin{bmatrix} \tilde{Q}_k & \tilde{M}_k \\ \tilde{M}_k^T & \tilde{R}_k \end{bmatrix} \begin{bmatrix} \Delta x_k \\ \Delta u_k \end{bmatrix} + \begin{bmatrix} q_k \\ r_k \end{bmatrix}^T \begin{bmatrix} \Delta x_k \\ \Delta u_k \end{bmatrix} \right\} \\
+\tfrac{1}{2}\Delta x_N^T \tilde{Q}_N \Delta x_N + q_N^T \Delta x_N + \sum_{k=1}^{N} \Xi(\eta_k + \Delta \eta_k)
\end{gathered}
$$

Hence, consider block-diagonal approximations:

- finite-difference approx to exact Lagrangian Hessian;
- partitioned quasi-Newton approximations;
- Hessian of the objective (i.e. ignore curvature of the constraints).

partitioned quasi-Newton

(Griewank-Toint 1982, Bock-Plitt 1984). Lagrangian can be separated as

$$\mathcal{L}(x,u,\lambda,\mu) = \mathcal{L}_0(u_0,\lambda_0,\mu_0) + \sum_{k=1}^{N-1} \mathcal{L}_k(x_k,u_k,\lambda_{k-1},\lambda_k,\mu_k) + \mathcal{L}_N(x_N,\lambda_{N-1}),$$

where

$$\mathcal{L}_k(x_k,u_k,\lambda_{k-1},\lambda_k,\mu_k) = \mathcal{C}(x_k,u_k) + \lambda_k^T F(x_k,u_k) - \lambda_{k-1}^T x_k + \mu_k^T(Du_k - d),$$

(ignoring state constraints). Exact kth block is

$$\begin{bmatrix} \tilde{Q}_k & \tilde{M}_k \\ \tilde{M}_k^T & \tilde{R}_k \end{bmatrix} = \begin{bmatrix} \dfrac{\partial^2 \mathcal{L}_k}{\partial x_k^2} & \left(\dfrac{\partial^2 \mathcal{L}_k}{\partial u_k \partial x_k}\right)^T \\ \dfrac{\partial^2 \mathcal{L}_k}{\partial u_k \partial x_k} & \dfrac{\partial^2 \mathcal{L}_k}{\partial u_k^2} \end{bmatrix}.$$

Quasi-Newton update for this block is based on step vector s_k and gradient change vector y_k defined as follows:

$$s_k = \begin{bmatrix} x_k^+ - x_k \\ u_k^+ - u_k \end{bmatrix},$$

$$\begin{aligned} y_k &= \begin{bmatrix} \frac{\partial}{\partial x_k}\mathcal{L}_k(x_k^+, u_k^+, \lambda_{k-1}, \lambda_k, \mu_k) - \frac{\partial}{\partial x_k}\mathcal{L}_k(x_k, u_k, \lambda_{k-1}, \lambda_k, \mu_k) \\ \frac{\partial}{\partial u_k}\mathcal{L}_k(x_k^+, u_k^+, \lambda_{k-1}, \lambda_k, \mu_k) - \frac{\partial}{\partial u_k}\mathcal{L}_k(x_k, u_k, \lambda_{k-1}, \lambda_k, \mu_k) \end{bmatrix} \\ &= \begin{bmatrix} \frac{\partial}{\partial x_k}\mathcal{C}(x_k^+, u_k^+) - \frac{\partial}{\partial x_k}\mathcal{C}(x_k, u_k) + (A_k(x_k^+, u_k^+) - A_k(x_k, u_k))^T\lambda_k \\ \frac{\partial}{\partial u_k}\mathcal{C}(x_k^+, u_k^+) - \frac{\partial}{\partial u_k}\mathcal{C}(x_k, u_k) + (B_k(x_k^+, u_k^+) - B_k(x_k, u_k))^T\lambda_k \end{bmatrix}, \end{aligned}$$

Use both BFGS (modified using Powell's method to retain positive definiteness) and SR1. If the latter, convexify the resulting matrix to allow the convex QP solver to be called.

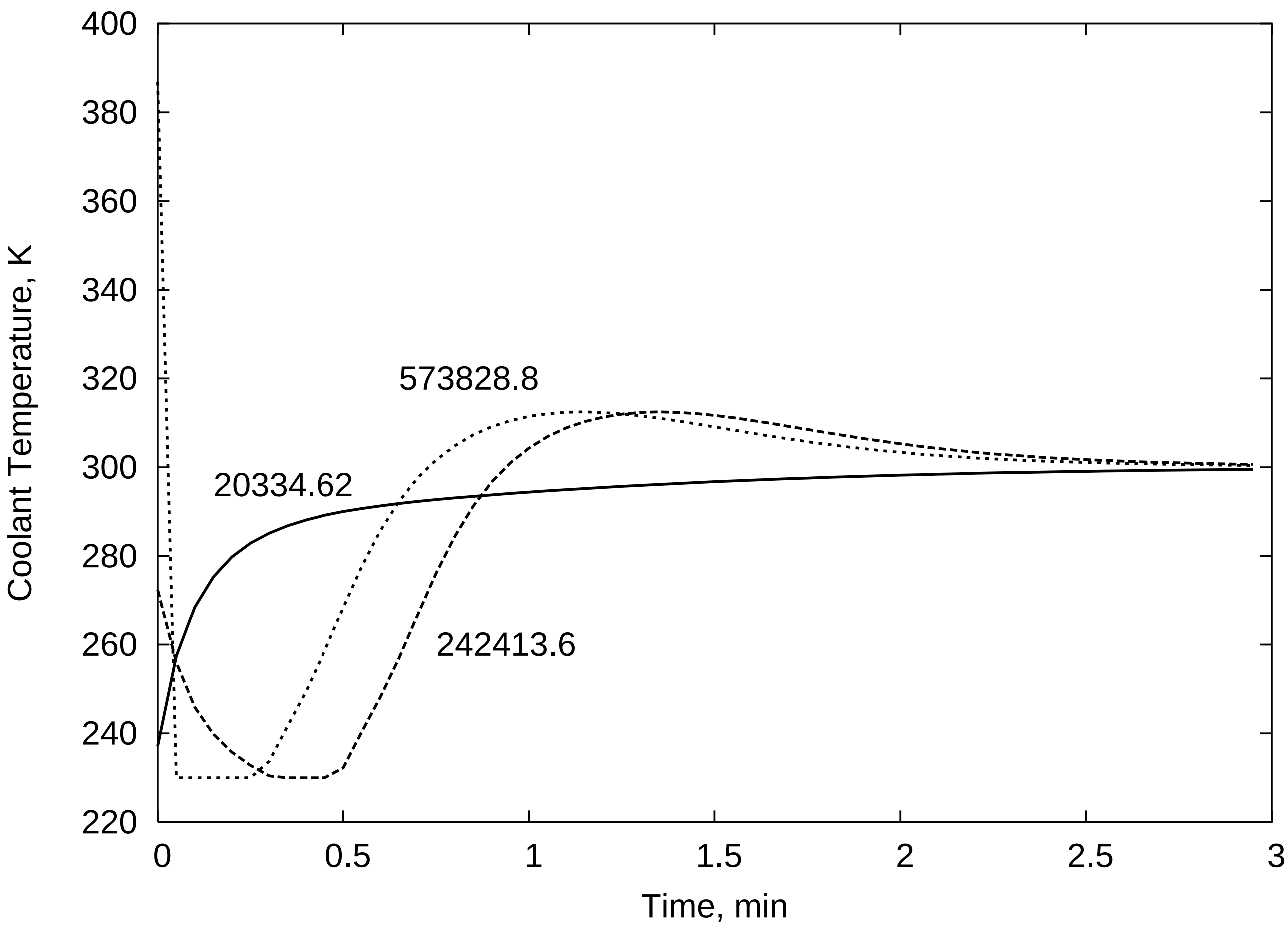
test problems

1. continuously stirred-tank reactor, involving an exothermic reaction, cooled via heat-exchange coil. 2 states, 1 control, $N = 60$. Open-loop unstable.
2. Mass spring damper. 2 states, 1 control, $N = 100$.
3. CSTR with 4 states, 2 controls, $N = 30$.
4. Pendulum on a cart. 2 states, 1 input (velocity of cart), $N = 30$.
5. copolymerization reaction / separation. 15 states, 3 inputs, $N = 20$.

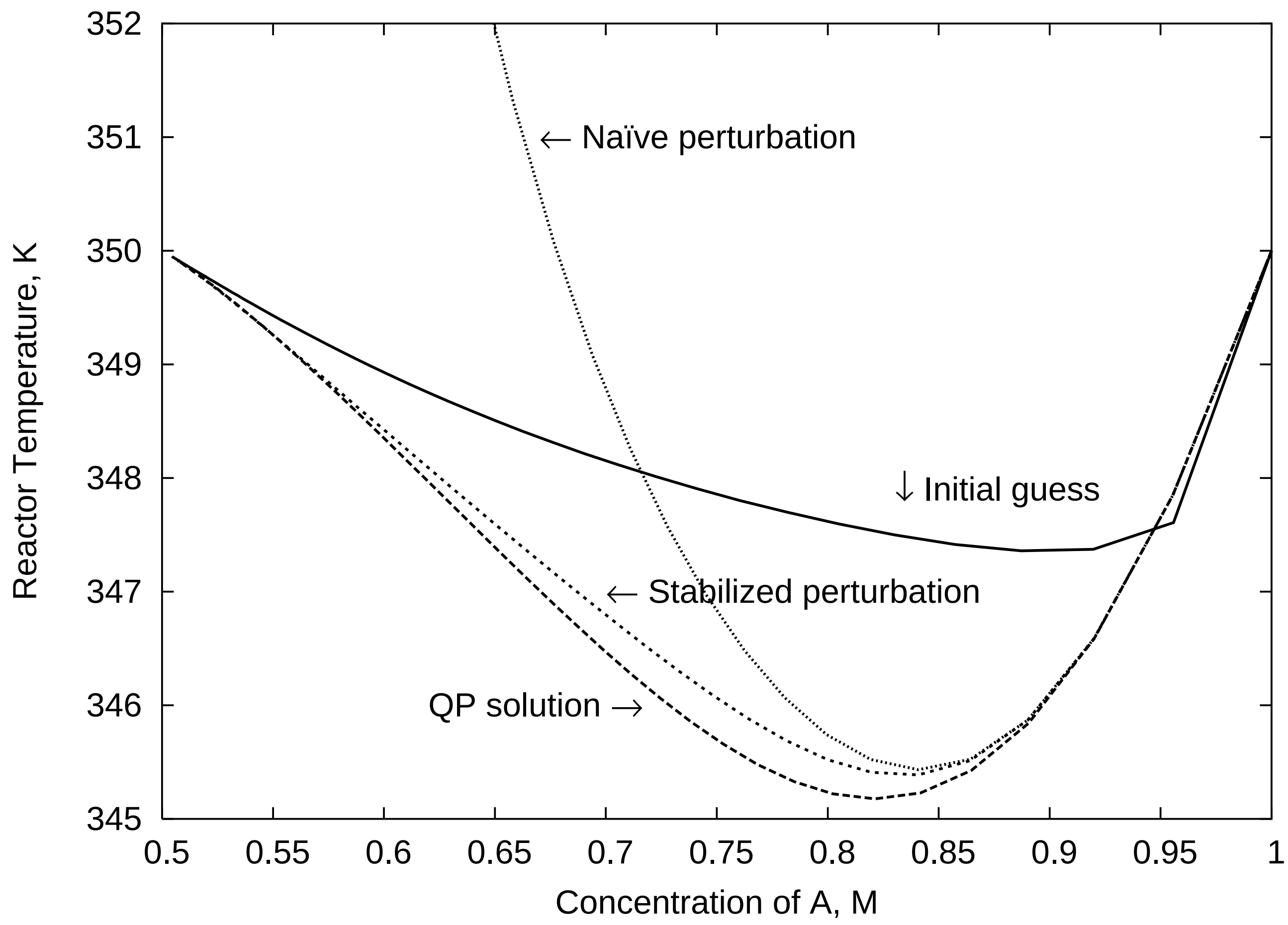
codes

- FP-SQP:
  - implemented in Octave;
  - LSODE integrates between timepoints;
  - DDASAC to calculate sensitivities (gradients).
- NPSOL
  - quasi-Newton; dense Hessian approx and linear algebra;
  - variants NPSOLu (eliminate x and η); NPSOLz (“simultaneous”).

Run on 1.2 GHz PC running Debian Linux.



Local solutions for Example 1. Note saturation in the two local minima.



effect of stabilization in feasibility perturbation: Example 1.

Method	Ex. 1	Ex. 2	Ex. 3	Ex. 4	Ex. 5
Finite-Difference Hessian	4 18.3	13 247.	4 39.2	5 78.6	3 197.
Objective Hessian	9 11.7	10 50.8	7 7.04	8 26.5	5 20.3
Partitioned BFGS	6 8.37	FAIL	8 9.92	8 26.7	4 16.5
Sparsified BFGS	7 10.1	10 52.6	7 7.45	8 27.2	4 17.2
Paritioned SR1	6 8.07	12 60.8	7 8.86	11 36.6	4 16.1
NPSOLu	FAIL	50 2280.	3 16.3	12 128.	FAIL
NPSOLz	23 6780.	$>$100 163000.	4 4870.	16 7840.	FAIL

comments on results

- runtimes not reliable: FP-SQP in Octave (interpreted); NPSOL doesn't use structure; sensitivity estimates expensive (affects finite-difference version)
- NPSOLu fails on Ex. 1, because of unstable elimination. Stabilized perturbation works well.
- Hessian-of-Objective and quasi-Newton strategies work well in general; hybrid method looks suitable.
- large number of NPSOL iterations suggests that we gain something from retaining feasibility.